Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges and arranged in a hierarchy with levels, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge wherein each bridge responds to configuration commands sent to its bridge ID, the method comprising:

   (a) selecting an initial bridge ID value;

   (~~a~~b) initially setting the bridge ID of all bridges to ~~a same common predetermined~~ the initial bridge ID value so that all bridges start with the same bridge ID;

   (c) configuring bridges on a hierarchical level so that only one bridge at a time responds to a configuration command sent to the initial bridge ID value;

   (d) ~~and walking the bus system to discover the bus topology and the bus bridges that form that topology by~~ repeatedly sending configuration commands and data to the ~~same predetermined~~ initial bridge ID value;

   (~~b~~e) assigning a unique bridge ID different from the ~~predetermined~~ initial bridge ID value to each ~~discovered~~ bridge that responds to the configuration commands and data; and

   (~~e~~f) entering information into internal registers and address bitmap of each ~~discovered~~ bridge that responds to the configuration commands and data to control the flow of information between bus segments.

2. (Currently Amended) The method of claim 1 wherein the bus topology is a tree configuration and ~~step (a) comprises~~ steps (e) and (f) comprise performing a recursive procedure that configures each branch of the tree.

3. (Currently Amended) The method of claim 1 wherein the bus system has an address space and wherein ~~step (a)~~ the method further comprises:

(~~a1~~g) probing the address space for slave devices.

4. (Currently Amended) The method of claim 3 wherein step ~~(a1)~~ (g) comprises:

~~(a1a)~~(g1) checking for a duplicate slave address when a slave device is located.

5. (Currently Amended) The method of claim 4 wherein step ~~(a1)~~ (g1) comprises:

(~~a1b~~g1a) inserting a slave address of a located slave device into a global address bitmap if the slave address is not a duplicate; and

(~~a1c~~g1b) inserting the slave address into a tunnel list if the slave address is a duplicate.

6. (Currently Amended) The method of claim 5 wherein step ~~(a1)~~ (g1) further comprises

(~~a1d~~g1c) repeatedly probing the address space for upstream bridges when no slave device is located.

7. (Currently Amended) The method of claim 6 wherein step ~~(b)~~ (e) comprises assigning a bridge ID value to each located upstream bridge.

8. (Currently Amended) The method of claim 7 wherein step ~~(a1)~~ (g1) further comprises

(~~a1e~~g1d) repeatedly probing for downstream bridges when no further upstream bridges are located in step ~~(a1d)~~(g1c).

9. (Currently Amended) The method of claim 8 wherein step ~~(b)~~ (e) comprises assigning a bridge ID value to each located downstream bridge.

10. (Currently Amended) The method of claim 9 wherein step ~~(c)~~ (f) comprises entering information into internal registers and address bitmap of at least one downstream bridge when no further downstream bridges are detected in step (~~a1e~~g1d).

11. (Currently Amended) The method of claim 1 further comprising:

(~~d~~g) walking the bus system to discover upstream bridges; and

(e~~h~~) entering information into internal registers and address bitmap of each discovered upstream bridge to control the flow of information between bus segments.

12. Cancelled.

13. (Currently Amended) The method of claim 1 wherein step ~~(b)~~ (c) comprises electrically connecting all bridges on the same hierarchical level together so that only one bridge at a time responds to configuration commands sent to the ~~same predetermined~~ initial bridge ID value.

14. (Original) The method of claim 13 wherein all bridges on the same hierarchical level are connected in a daisy chain configuration.

15. (Currently Amended) The method of claim 14 wherein a bridge in the daisy chain configuration enables the next bridge in the daisy chain configuration to respond

to the ~~same predetermined~~ initial bridge ID value when the bridge is assigned a bridge ID value other than the ~~same predetermined~~ initial bridge ID value.

16. (Currently Amended) The method of claim 1 wherein at least some of the bridges are bi-directional bridges comprised of two unidirectional bridges connected in parallel and wherein step ~~(b)~~ (e) comprises giving the two unidirectional bridges different bridge ID values.

17. (Currently Amended) The method of claim 1 further comprising:

([[f]]g) providing additional information to each bridge to enable the bridge to operate with a deterministic arbitration protocol.

18. (Currently Amended) Apparatus for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges and arranged in a hierarchy with levels, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge wherein each bridge responds to configuration commands sent to its bridge ID, the apparatus comprising:

a mechanism that selects an initial bridge ID value;

a configuration host that initially sets the bridge ID of all bridges to ~~a same common predetermined~~ the initial bridge ID value so that all bridges start with the same bridge ID;

a mechanism that configures bridges on a hierarchical level so that only one bridge at a time responds to a configuration command sent to the initial bridge ID value;

~~and then walks the bus system to discover the bus topology and the bus bridges that form that topology by~~ a mechanism that repeatedly ~~sending~~ sends configuration commands and data to the ~~same predetermined~~ initial bridge ID value;

a mechanism that assigns a unique bridge ID value different than the ~~same predetermined~~ initial bridge ID value to each ~~discovered~~ bridge that responds to the configuration commands and data; and

a mechanism that enters information into internal registers and address bitmap of each ~~discovered~~ bridge that responds to the configuration commands and data to control the flow of information between bus segments.

19. (Original) The apparatus of claim 18 wherein the bus topology is a tree configuration and the configuration host performs a recursive procedure that configures each branch of the tree.

20. (Original) The apparatus of claim 18 wherein the bus system has an address space and wherein the configuration host comprises a mechanism that probes the address space for slave devices.

21. (Original) The apparatus of claim 20 wherein the configuration host comprises a global address bitmap and a mechanism that uses the global address bitmap to check for a duplicate slave address when a slave device is located.

22. (Original) The apparatus of claim 21 wherein the configuration host comprises a mechanism that inserts a slave address of a located slave device into the global address bitmap if the slave address is not a duplicate; and inserts the slave address into a tunnel list if the slave address is a duplicate.

23. (Original) The apparatus of claim 22 wherein the configuration host comprises a mechanism that repeatedly probes the address space for upstream bridges when no slave device is located.

24. (Previously Presented) The apparatus of claim 23 wherein the bridge ID assigning mechanism comprises a mechanism that assigns a bridge ID value to each located upstream bridge.

25. (Original) The apparatus of claim 24 wherein the configuration host further comprises a mechanism that repeatedly probes for downstream bridges when no further upstream bridges are located by the upstream bridge locating apparatus.

26. (Previously Presented) The apparatus of claim 25 wherein the bridge ID assigning mechanism comprises a mechanism that assigns a bridge ID value to each located downstream bridge.

27. (Original) The apparatus of claim 26 wherein the information entering mechanism comprises a mechanism that enters information into internal registers and address bitmap of at least one downstream bridge when no further downstream bridges are detected by the downstream bridge locating mechanism.

28. (Original) The apparatus of claim 18 further comprising a mechanism that walks the bus system to discover upstream bridges and a mechanism that enters information into internal registers and address bitmap of each discovered upstream bridge to control the flow of information between bus segments.

29. (Canceled)

30. (Currently Amended) The apparatus of claim 18 wherein the ~~bridge ID assigning~~ mechanism that configures bridges comprises CFG IN/CFG OUT pins on each bridge wherein all bridges on the same hierarchical level have their CFG IN/CFG OUT pins connected together so that only one bridge at a time responds to the ~~same predetermined bridge~~ initial ID value.

31. (Original) The apparatus of claim 30 wherein the CFG IN/CFG OUT pins of all bridges on the same hierarchical level are connected in a daisy chain configuration.

32. (Currently Amended) The apparatus of claim 31 wherein a bridge in the daisy chain configuration enables the next bridge in the daisy chain configuration to respond to the ~~same predetermined~~ initial bridge ID value when the bridge is assigned a bridge ID value other than the ~~same predetermined~~ initial bridge ID value.

33. (Previously Presented) The apparatus of claim 18 wherein at least some of the bridges are bi-directional bridges comprised of two unidirectional bridges connected in parallel and wherein the bridge ID assigning mechanism comprises a mechanism that assigns the two unidirectional bridges different bridge ID values.

34. (Original) The apparatus of claim 18 further comprising a mechanism that provides additional information to each bridge to enable the bridge to operate with a deterministic arbitration protocol.

35. (Currently Amended) A computer program product for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges and arranged in a hierarchy with levels, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge wherein each bridge responds to configuration commands sent to its bridge ID, the computer program product comprising a computer usable medium having computer readable program code thereon, including:

program code that selects an initial bridge ID value;

program code that initially sets the bridge ID of all bridges to ~~a same common predetermined~~ the initial bridge ID value so that all bridges start with the same bridge ID;

program code that configures bridges on a hierarchical level so that only one bridge at a time responds to a configuration command sent to the initial bridge ID value;

~~and then walks the bus system to discover the bus topology and the bus bridges that form that topology by~~ program code that repeatedly ~~sending~~ sends configuration commands and data to the ~~same predetermined~~ initial bridge ID value;

program code that assigns a unique bridge ID value different than the ~~same predetermined~~ initial bridge ID value to each ~~discovered~~ bridge that responds to the configuration commands and data; and

program code that enters information into internal registers and address bitmap of each ~~discovered~~ bridge that responds to the configuration commands and data to control the flow of information between bus segments.

36. (Currently Amended) A computer data signal embodied in a carrier wave for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges and arranged in a hierarchy with levels, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge wherein each bridge responds to configuration commands sent to its bridge ID, the computer data signal comprising:

program code that selects an initial bridge ID value;

program code that initially sets the bridge ID of all bridges to ~~a same common predetermined~~ the initial bridge ID value so that all bridges start with the same bridge ID;

program code that configures bridges on a hierarchical level so that only one bridge at a time responds to a configuration command sent to the initial bridge ID value;

~~and then walks the bus system to discover the bus topology and the bus bridges that form that topology by~~ program code that repeatedly ~~sending~~ sends configuration commands and data to the ~~same predetermined~~ initial bridge ID value;

program code that assigns a unique bridge ID value different than the ~~same predetermined~~ initial bridge ID value to each ~~discovered~~ bridge that responds to the configuration commands and data; and

program code that enters information into internal registers and address bitmap of each ~~discovered~~ bridge that responds to the configuration commands and data to control the flow of information between bus segments.